In the Claims:

1. (Currently Amended) A processor for convolutional decoding, comprising:

a first processing unit for executing a first Viterbi instruction;

a second processing unit for executing a second Viterbi instruction;

a register comprising a plurality of ordered bit positions; and

update logic coupled to the register, the first processing unit and the second processing unit, the update logic configured to receive a first signal indicative of a result of a first add-compare-select instruction from the first processing unit and a second signal indicative of a result of a second add-compare-select instruction from the second processing unit, the update logic further, ~~and~~ configured to update the contents of the register dependent upon the first and second signals;

wherein, in the event that: (1) the first signal but not the second signal, or (2) the second signal but not the first signal, is received, the update logic is configured to shift the contents of the register such that one bit position is vacated and to update the vacated bit position dependent on the received signal; and

wherein in the event the first and second signals are received substantially simultaneously, the update logic is configured to shift the contents of the register 2 bit positions in order thereby vacating 2 consecutive bit positions, to update one of the vacated bit positions dependent upon the first signal, and to update the other vacated bit position dependent upon the second signal.

2. (Original) The processor as recited in claim 1, wherein the first add-compare-select instruction and the second add-compare-select instruction each specify two add operations, a compare operation, and a select operation.

3. (Original) The processor as recited in claim 2, wherein the first add-compare-select instruction and the second add-compare-select instruction each specify a first pair of source operands and a second pair of source operands, and wherein each of the add operations specifies summing one of the first pair of source operands and one of the second pair of source operands.

4. (Original) The processor as recited in claim 2, wherein the compare operation comprises comparing results of the two add operations.

5. (Original) The processor as recited in claim 2, wherein the select operation comprises producing a result dependent upon a result of the compare operation.

6. (Currently Amended) The processor as recited in claim 2, wherein the processor is configured ~~to execute~~ such that the first processing unit executes the first add-compare-select instruction and the second processing unit simultaneously executes the second add-compare-select instruction ~~simultaneously dependent~~ based upon a set of instruction grouping rules.

7. (Currently Amended) The processor as recited in claim 6, further comprising an instruction sequencing unit, ~~and an execution unit,~~ wherein the instruction sequencing unit is configured to issue decoded instructions to the ~~execution~~ first processing unit and the second processing unit for simultaneous execution dependent upon the set of instruction grouping rules.

8-9: (Canceled).

10. (Currently Amended) A processor for decoding convolutional code, comprising:

~~an execution unit configured to produce a first signal and a second signal substantially simultaneously when executing two add compare select instructions at the same time, wherein the first and second signals are indicative of results of the two add compare select instructions;~~

an instruction sequencing unit, said instruction sequencing unit configured to group two Viterbi instructions for simultaneous execution;

an arithmetic logic unit (ALU);

a multiply/accumulate unit (MAU);

said instruction sequencing unit issuing a first one of the two Viterbi instructions to the ALU for execution thereby;

the instruction sequencing unit issuing a second one of the two Viterbi instructions to the MAU for execution thereby;

the ALU executing a first add-compare-select (ACS) operation on the first Viterbi instruction generally simultaneous with the MAU executing a second ACS operation on the second Viterbi instruction;

a register comprising a plurality of ordered bit positions; and

update logic coupled to the register, the ALU and the MAU, the update logic configured ~~and the execution unit~~ to receive the first signal issued by the ALU and the second ~~signals and to update~~ signal issued by the MAU, the update logic updating the contents of the register ~~dependent upon the first and second signals; and~~ based upon the first signal issued by the ALU and the second signal issued by the MAU;

~~wherein in the event the first and second signals are received substantially simultaneously, the update logic is configured to shift the contents of the register 2 bit positions in order thereby vacating 2 consecutive bit positions of the register, to update one of the vacated bit positions dependent upon the first signal, and to update the other vacated bit position dependent upon the second signal.~~

wherein in the event that the first signal issued by the ALU and the second signal issued by the MAU are received substantially simultaneously, the update logic shifting the contents of the registry 2 bit positions in order to vacate 2 consecutive bit positions of the register, the update logic updating one of the vacated bit positions based upon the first signal issued by the ALU and updating the other vacated bit position based upon the second signal issued by the MAU.

11. (Original) The processor as recited in claim 10, wherein each of the two add-compare-select instructions specifies two add operations, a compare operation, and a select operation.

12. (Original) The processor as recited in claim 11, wherein each of the two add-compare-select instructions specifies a first pair of source operands and a second pair of source operands, and wherein each of the two add operations specifies summing one of the first pair of source operands and one of the second pair of source operands.

13. (Original) The processor as recited in claim 11, wherein the compare operation comprises comparing results of the two add operations.

14. (Original) The processor as recited in claim 11, wherein the select operation comprises producing a result dependent upon a result of the compare operation.

15. (Currently Amended) The processor as recited in claim 10, wherein the two add-compare-select instructions are dispatched to the ~~execution unit~~ ALU and the MAU for simultaneous execution dependent upon a set of instruction grouping rules.

16. (Currently Amended) The processor as recited in claim 15 further comprising an instruction sequencing unit configured to issue decoded instructions to the ~~execution unit~~ ALU and the MAU for simultaneous execution dependent upon the set of instruction grouping rules.

17-22: (Canceled).